

IN THE SPECIFICATION:

Please amend the specification as follows:

Page 5, lines 16-25, please amend as follows:

Composite Event When multiple events are combined using composition operators such as AND, OR, NOT, SEQUENCE, TIMES etc. to form a single event the resultant event is called as a composite event. The events taking part in the composite event can either be primitive events or composite events or a combination of these two. Composite event and its components are shown in the diagram ~~200~~ 100 of Fig. ~~2~~ 1 for a first Event 1 ~~205~~ 105 and a second Event 2 ~~210~~ 110. The two events ~~205~~ 105, ~~210~~ 110, which each comprise the actual event and an associated lifespan, are joined by a composition operator over an associated lifespan or time span as defined below.

Page 9, lines 15-25, please amend as follows:

Dynamic deployment of triggers

The triggers are intelligently deployed using the temporal conditions associated with the event part of the ECA rule. This temporal event can be in the form of the context of events, the lifespan of the rules as well as the lifespan of the composite events. Some representation of the ECA rule is taken as input. Based on these temporal conditions associated with the rule, the described techniques determine a minimal time during which the trigger can be defined in the application database. Fig. 4 presents the high-level view of an architectural framework 400 for dynamic deployment of triggers. The description of each of its components is given below. Related reference numerals are used for related components that appear in common with the schematic representation of Fig. 2.

Page 10, line 28 through page 11, line 5, please amend as follows:

Dynamic Trigger Deployment 418: When a notification is received from the temporal daemon 416, this module 418 identifies the triggers that need to be activated at that time. This can be done by querying the database 435 where the mapping between the event id and the trigger definition is stored. Once the module 418 has the event id, the module 418 defines the trigger on the underlying application database 420. An exception to this is in case of the Sequence operator. Consider an event Sequence (E_1, E_2, E_3). In this composite event, if the event E_1 has not occurred then the Dynamic Trigger Deployment module 418 does not deploy the trigger for event E_2 even if the module 418 receives a notification from the temporal daemon 416. The trigger is deployed only when the event E_1 occurs and if the lifespan of the event E_2 is active. To support this there is a need for an event monitor 425 that notifies the Dynamic Trigger deployment module 418 once the sub-events, taking part in a Sequence Composite event, fire.

Page 17, lines 9-19, please amend as follows:

Now consider the example event expression of Fig. 6. The Trigger Enforcement Engine 415 obtains the optimised tree that has a lifespan for all the events in the event expression. The temporal daemon 416 defines alarms for the start and end of the various events. Thus at time 15, the Dynamic Trigger Deployment module 418 receives an alarm to define trigger for event E_1 and E_3 . This module gets the trigger definition from the database ~~420~~ 435 and defines the trigger on the application database 420. The Dynamic Trigger Removal component 417 is invoked at the expiry of a lifespan. If at time 17, the Dynamic Trigger Removal component 417 finds that the event E_3 and the OR composite event has not occurred even once in its lifespan, then removes the temporal alarms for events E_4 and deletes the trigger defined for E_3 . This is done in the given situation the composite event AND is never true. The occurrence information of event E_1 and E_2 can be obtained by querying the Event Monitor 425.